

CS 494
Gray
Internet Draft
University
Intended status: IRC Class Project Specification
2018

Dalton

Portland State

November 25,

Internet Relay Chat Class Project

Status of this Memo

This document defines an experimental protocol for the Internet community.

Internet-Drafts

are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts. This document has been published for experimental implementation, and evaluation.

Abstract

This memo describes the communication protocol for an IRC-style client/server system for the Internetworking Protocols class at Portland State University.

Table of Contents

1. Introduction.....	1
2. Basic Information.....	2
3. Commands.....	2
4. Messages.....	3

1. Introduction

This specification describes a simple Internet Relay Chat (IRC) protocol by which clients can communicate with each other. This system employs a central server which “relays” messages that are sent to it to other connected users.

Users can join rooms, which are groups of users that are subscribed to the same message stream. Any message sent to that room is forwarded to all other users currently joined to that room.

2. Basic Information

All communication described in this protocol takes place over TCP/IP, with the server listening for connections on port 6429. Clients connect to this port and maintain this persistent connection to the server. The client can send messages and requests to the server over this open channel, and the server can reply via the same. This messaging protocol is inherently asynchronous - the client is free to send messages to the server at any time, and the server may asynchronously send messages back to the client.

Both the server and client may terminate the connection at any time for any reason. They may choose to send an error message to the other party informing them of the reason for connection termination.

The server may choose to allow only a finite number of users and rooms, depending on the implementation and resources of the host system.

3. Commands

LEAVE - This command allows a client to leave a room they are in. The server removes the room from the clients room list.

Usage: LEAVE <room>

EXIT - This command disconnects the client from the server. Upon receiving this command the server removes the client from any rooms they are in and then disconnects them from the server

Usage: EXIT

JOIN - This command allows a client to join a pre-existing room, or create one if the desired room does not exist. The server upon receiving this command searches for the specified room, if it is not found, it creates the room and places it in the clients room list.

Usage: JOIN <room>

HERE - This command lists all the users in a specified room. Upon receiving this

command the server searches all clients in that room and sends them to the requesting client.

Usage: HERE <room>

LIST - This command lists all of the rooms currently on the server. Upon receiving this command the server loops through its rooms list and sends them to the requesting client.

Usage: LIST

4. Messages

Messages between client and server are limited to 1024 bytes. When a client sends a message to the server the stream is parsed and the first word examined. If this word is not

one of the predefined commands it is assumed to be a message. If the client is currently in a room the server sends the message to all other users in the room. If the client is not in a room an error message is returned to the client telling them to join a room.

